# Spoofing Detection in Automatic Speaker Verification Systems Using DNN Classifiers and Dynamic Acoustic Features

Yu, Hong; Tan, Zheng Hua; Ma, Zhanyu; Martin, Rainer; Guo, Jun

*Published in:*
IEEE Transactions on Neural Networks and Learning Systems

*DOI (link to publication from Publisher):*
10.1109/TNNLS.2017.2771947

*Publication date:*
2018

*Document Version*
Accepted author manuscript, peer reviewed version

# Spoofing Detection in Automatic Speaker Verification Systems Using DNN Classifiers and Dynamic Acoustic Features

Hong Yu, Zheng-Hua Tan, *Senior Member, IEEE*, Zhanyu Ma, *Senior Member, IEEE*, Rainer Martin, *Fellow, IEEE*, and Jun Guo

***Abstract*— With the development of speech synthesis technology, automatic speaker verification (ASV) systems have encountered the serious challenge of spoofing attacks. In order to improve the security of ASV systems, many antispoofing countermeasures have been developed. In the front-end domain, much research has been conducted on finding effective features which can distinguish spoofed speech from genuine speech and the published results show that dynamic acoustic features work more effectively than static ones. In the back-end domain, Gaussian mixture model (GMM) and deep neural networks (DNNs) are the two most popular types of classifiers used for spoofing detection. The log-likelihood ratios (LLRs) generated by the difference of human and spoofing log-likelihoods are used as spoofing detection scores. In this paper, we train a five-layer DNN spoofing detection classifier using dynamic acoustic features and propose a novel, simple scoring method only using human log-likelihoods (HLLs) for spoofing detection. We mathematically prove that the new HLL scoring method is more suitable for the spoofing detection task than the classical LLR scoring method, especially when the spoofing speech is very similar to the human speech. We extensively investigate the performance of five different dynamic filter bank-based cepstral features and constant Q cepstral coefficients (CQCC) in conjunction with the DNN-HLL method. The experimental results show that, compared to the GMM-LLR method, the DNN-HLL method is able to significantly improve the spoofing detection accuracy. Compared with the CQCC-based GMM-LLR baseline, the proposed DNN-HLL model reduces the average equal error rate of all attack types to 0.045%, thus exceeding the performance of previously published approaches for the ASVspoof 2015 Challenge task. Fusing the CQCC-based DNN-HLL spoofing detection system with ASV systems, the false acceptance rate on spoofing attacks can be reduced significantly.**

***Index Terms*— Constant Q cepstral coefficients (CQCC), deep neural networks (DNNs) classifier, human log-likelihood (HLL), log-likelihood ratios (LLRs), speaker verification, spoofing detection.**

## I. Introduction

As a low cost and flexible biometric solution to person authentication, automatic speaker verification (ASV) has been widely applied in telephone or network access control systems, such as telephone banking or apartment security [1]. ASV technology aims to verify the registered speaker's identity by analyzing the speech signal and comparing it against pretrained models or templates [2]–[5].

However, with the popularity of social networks, more and more people share their audio or video recordings on social media platforms. An imposter can easily steal the voiceprint information of a target speaker through the Internet and use the stolen information to generate high quality speech signals similar to those of the target speaker, through voice conversion (VC) [6], [7] or speech synthesis (SS) [8], [9] techniques. The generated speech can then be used to attack ASV systems. This attack is called spoofing.

The mitigation of spoofing attacks has been the focus of many research works [10]–[12]. There are two general strategies to protect ASV systems. One is to develop a more robust ASV system which can resist spoofing attacks. Unfortunately, research shows that existing ASV systems are vulnerable to spoofing attacks [13]–[15]. The other popular strategy is to build a separate spoofing detection system which focuses only on distinguishing between natural and synthetic speech signals [12]. Fig. 1 depicts a block diagram of the latter approach where a separate spoofing detection system helps an ASV system to make more accurate decisions.

Due to the advantage of being easily incorporated into existing ASV systems, spoofing detection has become an important topic in antispoofing research [10], [14], [16], [17].

Spoofing detection can be realized either in the feature domain or the classifier domain. In the feature domain, many different types of features have been evaluated to find features

Manuscript received March 10, 2017; revised August 27, 2017; accepted November 1, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 61773071 and Grant 61628301, in part by the Beijing National Science Foundation under Grant 4162044, in part by the Beijing Nova Program under Grant Z171100001117049, in part by the Chinese 111 program of Advanced Intelligence, OCTAVE-Objective Control for TAlker VErification funded by the Research Executive Agency of the European Commission through its framework program Horizon 2020 under Project 647850. The work of R. Martin was supported by the German Science Foundation within the framework of the Research Unit FOR2457 "Acoustic Sensor Networks." *(Corresponding author: Zhanyu Ma.)*

H. Yu, Z. Ma, and J. Guo are with the Pattern Recognition and Intelligent System Laboratory, School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: mazhanyu@bupt.edu.cn).

Z.-H. Tan is with the Department of Electronic Systems, Aalborg University, 9100 Aalborg, Denmark.

R. Martin is with the Institute of Communication Acoustics, Ruhr-Universität, 44801 Bochum, Germany.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2017.2771947

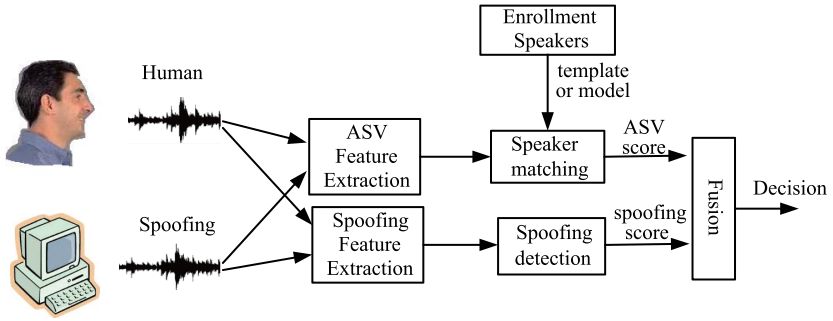



Fig. 1. Illustration of a spoofing detection system used in an ASV system.

which can distinguish the human voice from spoofing speech more effectively. Magnitude-based features, such as the log-magnitude spectrum (LMS) and the residual LMS, were tested in [18]. Phase-based features, e.g., relative phase shift, group delay (GD), modified GD (MGD), baseband phase difference, and cosine normalized phase features, were investigated in [17]–[19]. Spectrum and phase-based cepstral coefficients features, such as linear-frequency cepstral coefficients (LFCC), cochlear filter cepstral coefficients, mel-frequency cepstral coefficients (MFCC), cosine-normalized phase-based cepstral coefficients, and MGD filter-bank cepstral coefficients were studied in [17] and [20]–[24]. Other kinds of features, such as local binary patterns, pitch pattern features, utterance level i-Vectors [3], and modulation features were also discussed in [19] and [25]–[27].

The results published in [20] and [28] show that dynamic acoustic features are more effective than static features in the spoofing detection task. Moreover, the high frequency regions of speech play a more important role than the low frequency regions.

Inspired by the successful application of deep neural networks (DNNs) in feature extraction [29]–[31], DNN bottleneck features generated by the hidden layer of a DNN have also been used in the spoofing detection task. In [32], the utterance-level spoofing-vectors were generated by computing the mean values of DNN bottleneck features that were produced by filter bank features. Frame-level DNN bottleneck features derived from dynamic acoustic features, such as dynamic filter-bank, dynamic MFCC, and dynamic linear prediction cepstral coefficients, were discussed in [33]. Recurrent neural networks, such as the unidirectional long-short-term memory network and the bidirectional long-short-term memory network, were also used to produce sequence-level bottleneck features in [34].

Summarizing the published experimental results on the ASVspoofing 2015 database [35], DNN features perform better at detecting known spoofing attacks, but worse on detecting unknown attacks when using Gaussian mixture model (GMM)-based classifier. Average equal error rates (EERs) of all the attacks are in the range from 1% to 3%. Dynamic spectrum-based cepstral coefficients features, such as LFCC, and the newly published constant-Q cepstral coefficients (CQCC) [36] perform better than other features, and average EERs of these features can be reduced to less than 0.9%.

In the classifier domain, some classical classifiers, e.g., GMM-based log-likelihood ratios (GMM-LLRs) [20], linear discriminant analysis (LDA) [37], probabilistic LDA (PLDA) [38], binary/one-class support vector machine (SVM) [39], and DNN were all applied to the spoofing detection task.

In the GMM-LLR method, two separated GMMs are trained on the corresponding genuine and spoofing features. Spoofing detection scores are generated by computing the LLR of test speech on the two models [17]–[20]. When using the PLDA/LDA classifiers, two feature vectors, such as mean i-Vectors or statistical features, which stand for the genuine class and the spoofing class, should be generated first. Then spoofing detection scores are produced by evaluating the similarities between the test feature vectors and the genuine/spoofing feature vectors [25], [34]. In binary/one-class SVM, the output score of the classifiers can be used for spoofing detection directly [18], [20], [21], [34]. When a DNN classifier is applied, human speech and different spoofing methods are used as training labels and the posterior probabilities gotten from all nodes in the output layer are used to derive spoofing detection scores [18], [32]–[34].

Published results show that GMM-LLR classifiers trained using dynamic acoustic features work better than DNN and other classifiers [18]–[20], [34], so we select GMM-LLR classifiers as the baseline.

This paper focuses on studying DNN classifiers trained on dynamic acoustic features. Specifically, we propose a simple novel scoring method for DNN classifiers, which can make DNN classifiers perform much better than the baseline GMM-LLR classifier. The main contributions of this paper include the following.

1) We propose a new human log-likelihood (HLL)-based scoring method which uses only the output of the human node.
2) We mathematically prove that the HLL scoring method is more suitable for the spoofing detection task than the classical LLR method, especially, for detecting spoofing speech which is very similar to the human speech.
3) We investigate the performance of DNN-HLL models trained by five different dynamic spectrum-based cepstral coefficients and CQCC features. The CQCC-based DNN-HLL model works best on the ASVspoofing 2015 database. The average EER of all 10 attacks can be

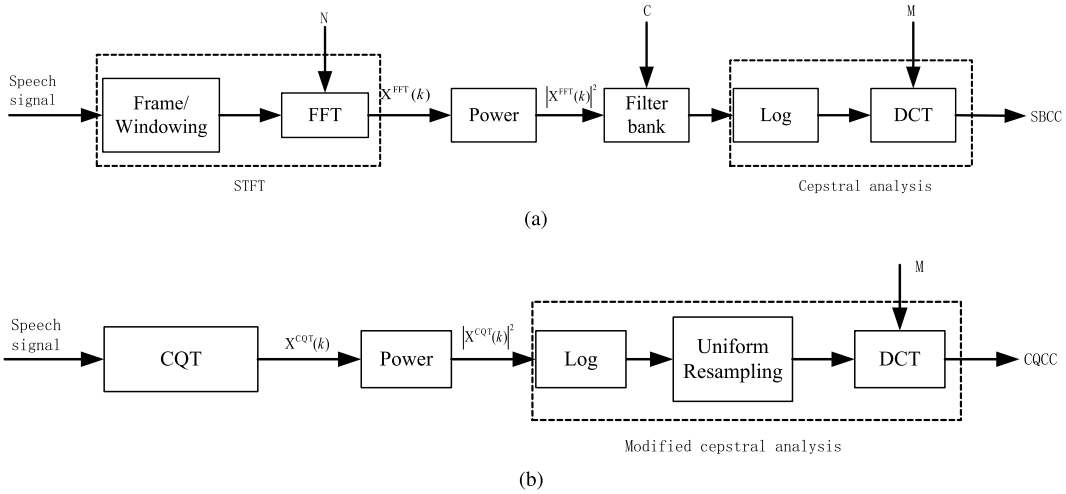



Fig. 2. Processing flow of computing SBCC and CQCC features, where $N$, $C$, and $M$ stand for the number of FFT points, the number of filter bank channels, and the number of cepstral coefficients, respectively. (a) Block diagram for SBCC extraction. (b) Block diagram for CQCC extraction.

reduced to 0.045%, which, to our knowledge, is the best performance among all the published results.

4) We integrate the CQCC-based DNN-HLL spoofing detection classifier with GMM- universal background model (UBM) and i-Vector-based ASV systems. The false acceptance rate (FAR) of untrained spoofing attacks can be reduced significantly.

The remaining part of this paper is organized as follows. In Section II, we introduce the dynamic acoustic features used for training spoofing detection classifiers. The detailed DNN structure for spoofing detection and the different scoring methods are described in Section III. In this section, we also mathematically prove that the HLL scoring method, using only the HLL, works better than the classical LLR methods on the spoofing detection task. The experimental results of spoofing detection and the effect of integrating spoofing detection and ASV are discussed in Section IV. We conclude this paper in Section V.

## II. Dynamic Features

Since frame level spectrum-based cepstral coefficients (SBCC) and CQCC work better than other features, such as phase features and utterance level features, this paper uses dynamic SBCC and CQCC to evaluate the performance of DNN classifiers. SBCC features, e.g., MFCC and LFCC, have been widely used in many speech processing tasks. They can be created with the procedure shown in Fig. 2(a). First, the speech signal is segmented into short-time frames with overlapping windows. Second, the power spectra $|X^{\mathrm{FFT}}(k)|^2$ are generated by a frame-wise $N$-point fast Fourier transform (FFT). Third, the power spectrum is integrated using an overlapping band-limited filter bank with $C$ channels to generate filter bank features. Finally, after logarithmic compression and discrete cosine transform (DCT) on the filter bank features, $M$ coefficients are selected as the SBCC feature.

We evaluate five different SBCC features generated by different filter banks as shown in Fig. 3. LFCC and MFCC are generated by linear frequency and mel-frequency rectangular filter banks (MFB), respectively. Gammatone filter bank (GFB) cepstral coefficients (GFCC), which have been successfully used in speech recognition tasks [40]–[42], are produced by the GFB which has mel-scale center frequencies and bandwidths in proportion to the equivalent rectangular bandwidth scale.

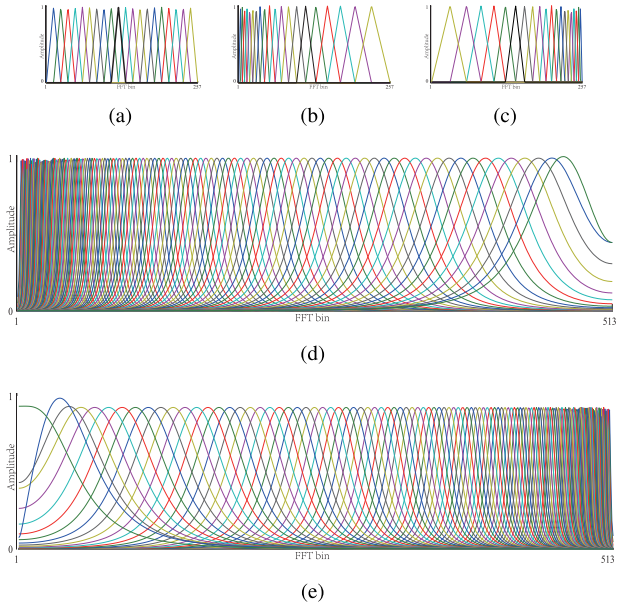



Fig. 3. Filter banks used for SBCC extraction. (a) TFB. (b) MFB. (c) IMFB. (d) GFB. (e) IGFB.

The experimental results published in [20] and [28] show that the high-frequency spectrum of speech is more effective for the detection of synthetic speech. Therefore, we also evaluate *inverted* MFCC (IMFCC) and *inverted* Gammatone filter bank cepstral coefficients (IGFCC) which are generated by the *inverted* MFB and GFB. These *inverted* filter banks have sparser spacing of filters in the low-frequency region and denser spacing in the higher frequency region

TABLE I
DESCRIPTION OF SBCC FEATURES USED IN THIS PAPER

| Feature Name | FFT ($N$) | Channel ($C$) | Coef. ($M$) | Filter bank |
|---|---|---|---|---|
| LFCC | 512 | 20 | 20 | TFB |
| MFCC | 512 | 20 | 20 | MFB |
| IMFCC | 512 | 20 | 20 | IMFB |
| GFCC | 1024 | 128 | 20 | GFB |
| IGFCC | 1024 | 128 | 20 | IGFB |

emphasizing the importance of the high frequency components [Fig. 3(c) and (e)].

The details of the SBCC features used in this paper are described in Table I. Following the suggestion in [20] and [42], the speech signals were segmented into frames with 20-ms window length and 10-ms step size. Pre-emphasis and a Hamming window were applied on the frames before the spectrum computation. The works in [20] showed that all the frames of speech are useful for spoofing detection, so we did not apply any voice activity detection method. When extracting LFCC, MFCC, and IMFCC, the channel number $C$ is set to 20 and FFT length $N$ is set to 512. When producing GFCC and IGFCC, $C$ and $N$ are set to 128 and 1024, respectively. The number of coefficients $M$ of all the SBCC features is set to 20 (including the 0th coefficient).

As shown in Fig. 2(b), CQCC features are generated from the constant-Q transform (CQT) [43]. Unlike the short-time Fourier transform used in the SBCC feature extraction, which has both uniform time and frequency resolutions, CQT can capture more detailed time-frequency information of speech signals. Higher frequency resolution is provided for lower frequencies and higher time resolution is applied on higher frequencies. Since the CQT frequency scale is geometrically spaced, a cepstral analysis method cannot be directly applied on the power CQT features $|X^{\mathrm{CQT}}(k)|^2$. A modified cepstral analysis method is taken by using a spline interpolation method to resample the geometric scale to a uniform grid [36], and then, after DCT, $M$ CQCC coefficients can be produced.

CQCC features are extracted by the code supplied from http://audio.eurecom.fr/content/software. The number of bins per octave is set to be 96 and resampling is applied with a sampling period of 16. The coefficient number $M$ is also set to 20 (including the 0th coefficient).

## III. CLASSIERS AND SCORING METHODS

This section presents the baseline GMM-LLR classifier and the proposed DNN-HLL method together with a mathematical proof that the HLL scoring method works better than the classical LLR method on the spoofing detection task.

### A. GMM-LLR Classifiers

The GMM-UBM approach which has been widely used in ASV systems [44] can easily be extended to spoofing detection. As shown in Fig. 4, human/spoofing GMM models are trained by spoofing detection features extracted from human/spoofing utterances, respectively, with an expectation maximization algorithm. All spoofed speech samples are fused together to train one spoofing GMM.

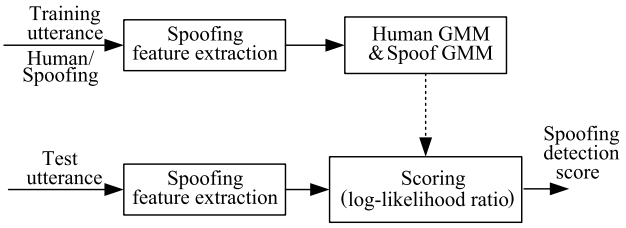



Fig. 4. Block diagram of a GMM-LLR-based spoofing detection system.

In the test phase, the spoofing detection features extracted from the test utterances are scored against the human and spoof GMMs. The LLR is used as spoofing detection score, calculated by the following equation:

$$\mathbf{S_{GMM}}(\mathbf{X}) = \frac{1}{T}\sum_{i=1}^{T}\{\log P(X_i|\lambda_{\mathrm{human}}) - \log P(X_i|\lambda_{\mathrm{spoof}})\} \tag{1}$$

where $\mathbf{X}$ denotes spoofing detection feature vectors with $T$ frames, $\lambda_{\mathrm{human}}$ and $\lambda_{\mathrm{spoof}}$ are the GMM parameters of human and spoofing models, respectively.

### B. DNN-LLR Classifiers

Spoofing-discriminant DNNs with five hidden layers are used to distinguish the human/spoofing speech in this paper. As shown in Fig. 5, each of the hidden layers has 2048 nodes with a sigmoid activation function. The number of nodes of the softmax output layer is $K+1$, corresponding to human speech and $K$ different spoofing attacks. Batch normalized super vectors $F_i$ which are composed of $n$ successive dynamic acoustic features described in Section II are used for DNN training, that is

$$F_i = \left[X_{i-\frac{n-1}{2}}, \ldots, X_i, \ldots, X_{i+\frac{n-1}{2}}\right]. \tag{2}$$

With the help of the computational network toolkit [45], the DNN is built and trained with stochastic gradient descent methods. The cross entropy function is selected as the cost function and the maximum training epoch is chosen as 120. The mini-batch size is set as 128.

The output of a trained DNN can be used for a spoofing decision directly or to generate spoofing detection scores. We use $P(h|F_i)$ and $P(s_k|F_i)$ $(k \in (1, \ldots, K))$ to stand for the output of the human node and the $k$th spoofing node, which also represent the posterior probability that the $i$th input super vectors $F_i$ belongs to human speech or the $k$th spoofing method.

As shown in (3), when the DNN is used as a decision device, by counting how many frames are more similar to a human frame, we can determine whether the input utterance is spoofed speech, that is

$$\mathrm{Decision}(\mathbf{F}) = \begin{cases} 1, & \text{count } (P(h|F_i) > 0.5) > 0.5T \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

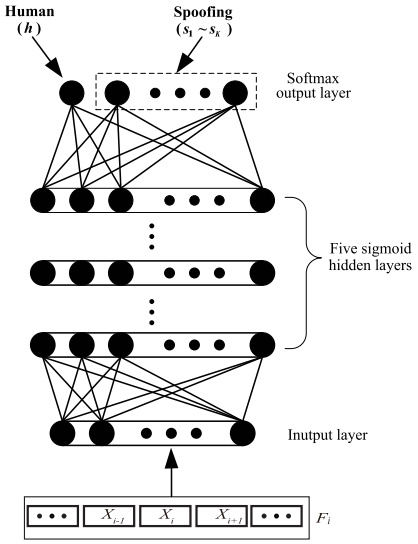



Fig. 5. Spoofing-discriminant DNN.

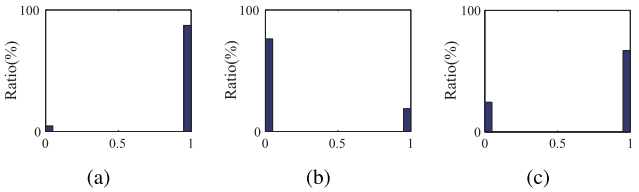



Fig. 6. Distribution of the human node output. (a) Human. (b) S1–S9. (c) S10.

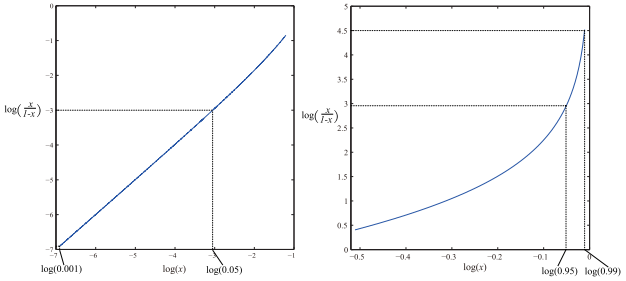



Fig. 7. Relationship between $\log(x)$ and $\log(x/(1-x))$.

where $\mathbf{F}$ stands for super vectors of the test utterance with $T$ frames.

When a DNN is used to score the input utterance, the posterior probabilities given by the output layer can be used to compute the spoofing detection score. Similar to the LLR scoring method used in the GMM classifier, we also use the difference between human and spoofing log-likelihoods to score the input utterance, as shown in

$$S1_{\text{DNN}}(\mathbf{F}) = \frac{1}{T}\sum_{i=1}^{T}\left\{\log[P(h|F_i)] - \log\left[\sum_{k=1}^{K} P(s_k|F_i)\right]\right\} \tag{4}$$

$$S2_{\text{DNN}}(\mathbf{F}) = \frac{1}{T}\sum_{i=1}^{T}\{\log[P(h|F_i)] - \log[\max(P(s_k|F_i))]\} \tag{5}$$

The spoofing log-likelihood for (4) and (5) is calculated by the sum or the maximum value of $K$ spoofing nodes, respectively.

The performance of spoofing detection will be evaluated using EER. In practical applications, a development data set will be used to find an EER threshold, and when the spoofing detection score is bigger than this threshold, the input utterance will be accepted as a human utterance, otherwise it will be rejected as a spoofing one.

### *C. DNN-HLL Classifier*

We first evaluated the performance of the MFCC-based DNN-LLR classifier on the ASVspoof2015 database which includes 10 different spoofing methods, S1–S10. The experimental results show that, on the S1–S9 attacks, the DNN classifier can relatively easily distinguish spoofed speech from human speech. However, under the S10 attack, the classifier performs quite badly (more details will be introduced in Section IV).

To understand this behavior, we investigate output values of the DNN classifier. Let $x$ denote the output of the human node. The distribution of $x$ is shown in Fig. 6.

It is observed that, most of the $x$ values of human speech fall into a high value interval near 1 [Fig. 6(a)]. For easily detected spoofing utterances (S1–S9), most of the $x$ values lie in a low value interval near 0 [Fig. 6(b)]. While for some spoofing utterances which are very difficult to detect (S10), as shown in Fig. 6(c), most of $x$ still fall into the high value range. If we use (3) to make the spoofing decision, these utterances will be wrongly judged as human speech. When using the LLR scoring method, the spoofing detection scores are calculated by $\log(x/(1-x))$. As shown in Fig. 7, in the low value interval, e.g., $x \in [0.001, 0.05]$, $\log(x)$, and $\log(x/(1-x))$ have a similar distribution. However, in high value interval, e.g., $x \in [0.95, 0.99]$, the transformation from $\log(x)$ to $\log(x/(1-x))$ make the distribution become scattered, which will increase variance value of spoofing detection scores. Scattered scores are not suitable for the classification task. In the LLR method, a wrong posterior probability $x$ of one frame, that fall into the high value interval, will cause a strong positive bias and affect the average log-likelihood score heavily.

This encourages us to use the log-likelihood of $x$ (human node outputs), to compute spoofing scores instead of the LLRs.

We introduce a novel, simple HLL scoring method as

$$S3_{\text{DNN}}(\mathbf{F}) = \frac{1}{T}\sum_{i=1}^{T}\log(P(h|F_i)). \tag{6}$$

The HLL score emphasizes frames which achieve a low "human" score near 0. So, once we have a certain number of low "human" scores in an utterance, the whole utterance will be scored as spoofed. This is different from LLR where low and high "human" scores are given asymmetric score.

In the rest of this section, we mathematically prove that the HLL scoring method is more suitable for the spoofing detection task than the LLR method.

We rewrite the HLL scoring method of (6) as

$$S_{\mathrm{HLL}} = \frac{1}{T}\sum_{i=1}^{T}\log(x_i) \tag{7}$$

where $x_i$ stands for the probability that the $i$th frame belongs to human utterance.

Equations (4) and (5), $\sum_{k=1}^{K} P(s_k|F_i)$ and $\max(P(s_k|F_i))$ both stand for the probability that the $i$th frame belongs to spoofing utterance, so we approximate them as $1 - x_i$. Equations (4) and (5) can then be rewritten as

$$S_{\mathrm{LLR}} = \frac{1}{T}\sum_{i=1}^{T}[\log(x_i) - \log(1 - x_i)]. \tag{8}$$

Following the distribution of $x$ shown in Fig. 6, the probability density functions (PDFs) of $x$ can be assumed to have a piecewise uniform distribution:

$$f(x) = \begin{cases} \dfrac{\alpha}{\beta_1}, & x \in [1-\beta_1, 1] \\ \dfrac{1-\alpha}{\beta_2}, & x \in [0, \beta_2] \end{cases} \tag{9}$$

where $0 < \alpha < 1$ is the probability that $x$ falls into the high value interval. $\beta_1$ and $\beta_2 \in (0, 0.5)$ stand for the width of low/high value intervals, respectively.

Through variable substitutions, the corresponding PDFs of $y_1 = \log(x)$ and $y_2 = \log(x) - \log(1-x)$ can be calculated as

$$f(y_1) = \begin{cases} \dfrac{1-\alpha}{\beta_1} e^{y_1}, & y_1 \in [-\infty, \log\beta_1] \\ \dfrac{\alpha}{\beta_2} e^{y_1}, & y_1 \in [\log(1-\beta_2), 0] \end{cases} \tag{10}$$

$$f(y_2) = \begin{cases} \dfrac{1-\alpha}{\beta_1}\dfrac{e^{y_2}}{(1+e^{y_2})^2}, & y_2 \in \left[-\infty, \log\left(\dfrac{\beta_1}{1-\beta_1}\right)\right] \\ \dfrac{\alpha}{\beta_2}\dfrac{e^{y_2}}{(1+e^{y_2})^2}, & y_2 \in \left[\log\left(\dfrac{1-\beta_2}{\beta_2}\right), +\infty\right]. \end{cases} \tag{11}$$

We assume $x_i$ meets the independent and identical distribution condition. Following the central limit theorem, the distributions of $S_{\mathrm{HLL}}$ and $S_{\mathrm{LLR}}$ can be approximated by normal distributions and their mean values and standard deviations can be computed by the expectations and standard deviations of $y_1$ and $y_2$ as follows:

$$m\_S_{\mathrm{HLL}} = E[y_1] \tag{12}$$

$$\sigma\_S_{\mathrm{HLL}} = \sqrt{\left(E\left[y_1^2\right] - E[y_1]^2\right)/T} \tag{13}$$

$$m\_S_{\mathrm{LLR}} = E[y_2] \tag{14}$$

$$\sigma\_S_{\mathrm{LLR}} = \sqrt{\left(E\left[y_2^2\right] - E[y_2]^2\right)/T}. \tag{15}$$

The distributions of human and spoofing scores are illuminated in Fig. 8, where $m_h$, $\sigma_h$ and $m_s$, $\sigma_s$ are mean and standard deviations of human scores and spoofing scores, respectively.

In scoring-based spoofing detection tasks, EER is used to evaluate the performance of different spoofing detection methods. Let $\mathrm{FRR}(\theta)$ and $\mathrm{FAR}(\theta)$ denote the false rejection and FARs at threshold $\theta$

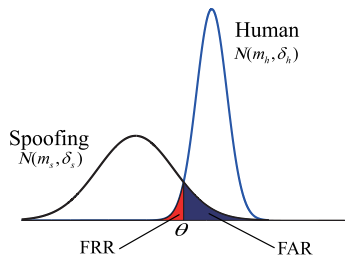



Fig. 8. Distributions of human and spoofing scores.

$$\mathrm{FAR}(\theta) = \frac{\text{count(spoof trials with score} > \theta)}{\text{total spoof trials}} \tag{16}$$

$$\mathrm{FRR}(\theta) = \frac{\text{count(human trials with score} < \theta)}{\text{total human trials}}. \tag{17}$$

$\mathrm{FRR}(\theta)$ and $\mathrm{FAR}(\theta)$ are monotonically decreasing and increasing functions of $\theta$. The EER corresponds to the threshold $\theta_{\mathrm{EER}}$ at which two detection error rates are equal. Because $\theta_{\mathrm{EER}}$ do not have a closed form solution in the simulated condition, we select the intersection of two Gaussian curves as threshold $\theta$, as shown in (18) that is shown at the bottom of the next page, which is slightly different from the $\theta_{\mathrm{EER}}$ threshold. Instead of EER, $\mathrm{FRR}(\theta)$, and $\mathrm{FAR}(\theta)$ are used to evaluate the performance of two scoring methods.

$\mathrm{FRR}(\theta)$ and $\mathrm{FAR}(\theta)$ can be estimated as

$$\mathrm{FRR}(\theta) = \mathrm{CDF}(\theta|m_h, \sigma_h) \tag{19}$$

$$\mathrm{FAR}(\theta) = 1 - CDF(\theta|m_s, \sigma_s) \tag{20}$$

where $CDF$ denotes the cumulative distribution function of the normal distribution.

Here we set $\beta_1 = \beta_2 = 0.05$, $T = 100$. Mean values and standard variances of $S_{\mathrm{HLL}}$ and $S_{\mathrm{LLR}}$ can be calculated as[1]

$$m\_S_{\mathrm{HLL}} \approx 3.97\alpha - 4 \tag{21}$$

$$\delta\_S_{\mathrm{HLL}} \approx \sqrt{-(3.97\alpha - 4)^2 - 16.96\alpha + 16.97}/10 \tag{22}$$

$$m\_S_{\mathrm{LLR}} \approx 7.94\alpha - 3.97 \tag{23}$$

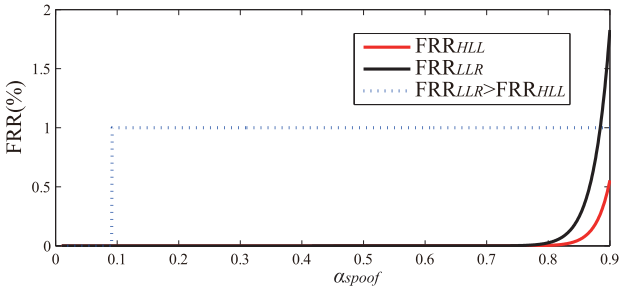
$$\delta\_S_{\mathrm{LLR}} \approx \sqrt{-(7.94\alpha - 3.97)^2 + 16.79}/10. \tag{24}$$

In order to simulate the distribution shown in Fig. 6(a), we set $\alpha$ of human scores as a constant value, $\alpha_{\mathrm{human}} = 0.99$. It means that, for human speech, 99% output values of human node fall into the high value area. The $\alpha$ value of spoofing scores, $\alpha_{\mathrm{spoof}}$, changes from 0.01 to 0.9. By decreasing $\alpha_{\mathrm{spoof}}$, the difference between human and spoofing speech grows. When $\alpha_{\mathrm{spoof}}$ is lower than 0.5, the spoofing attack can be detected relatively easily [Fig. 6(b)]. When $\alpha_{\mathrm{spoof}}$ is larger than 0.5 and approaches to 0.9, the spoofing attack is difficult to be detected [Fig. 6(c)].

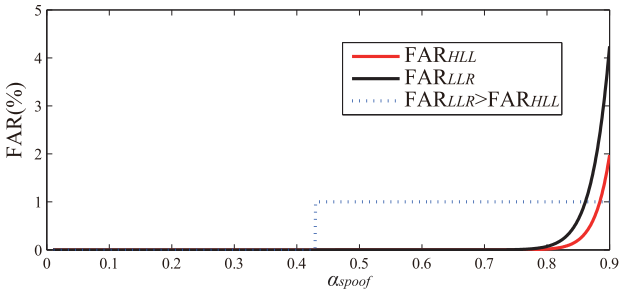
Using (18), (20), (22), and (24) FAR and false rejection rate (FRR) computed by LLR and HLL scoring methods are shown in Fig. 9.

The dotted lines label the areas where the FAR or FRR generated by the LLR method is larger than that computed by the HLL method.

[1]Computed by the Maple software, https://www.maplesoft.com/.




(a)




(b)

Fig. 9. Comparison of FRR and FAR generated by different scoring methods. (a) FRR for $\alpha_{\text{human}} = 0.99$. (b) FAR for $\alpha_{\text{human}} = 0.99$.

TABLE II
DESCRIPTION OF ASVSPOOF 2015 DATABASE

| Subsets | Speaker | | Utterances | |
|---|---|---|---|---|
| | Male | Female | Genuine | Spoofed |
| Training | 10 | 15 | 3750 | 12625 |
| Development | 15 | 20 | 3497 | 49875 |
| Evaluation | 20 | 26 | 9404 | 184000 |

The results show that when $\alpha_{\text{spoof}}$ is bigger than 0.5, FRR and FAR are still very small, less than 4%, which means most of the spoofed speech is detected correctly. It proves that the scoring method works better than the decision method described in (3).

Comparing the two scoring methods, when $\alpha_{\text{spoof}}$ has a large value, which means the spoofing speech is very similar to the human speech, the HLL scoring method performs better than LLR. Only when $\alpha_{\text{spoof}}$ is small enough, less than 0.45, the LLR scoring method can work better than the HLL method. However, in that case, the FRR and FAR of two scoring methods all tend to 0%. Thus, generally speaking, the scoring method, which only uses the HLL, is more suitable for spoofing detection task, especially for detecting high quality spoofed speech which is very similar to human speech.

## IV. EXPERIMENTS

### A. Database

The performance of spoofing detection using different features and classifiers is evaluated on the ASVspoof 2015 database [35]. As shown in Table II, the database includes three sub data sets without target speaker overlap: the training set, the development set, and the evaluation set. We used the training set for training spoofing detection classifiers. The development set and the evaluation set were used for testing. The training set and development set are attacked by the same five spoofing methods, where S1, S2, and S5 belong to VC category and S3, S4 belong to SS category. Regarding the evaluation set, besides the five known spoofing methods, there are another five unknown methods, where S6–S9 are VC methods and S10 is an SS method.

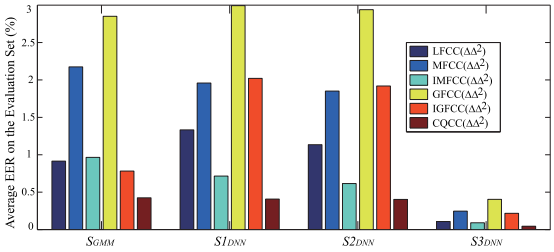



Fig. 10. Average EERs of different spoofing features and classifiers on the evaluation set.

Many recently released results show that the existing spoofing detection systems perform much worse on detecting the S10 attack, which is a unit-selection-based attack. The unit-selection-based attack is produced by concatenating the time-domain waveform directly without vocoding and feature extraction techniques, which does not carry much artifacts information from the perspective of feature representations [11].

The experimental results in the following section will show that the proposed DNN-HLL classifier is able to work very well on detecting the S10 attack as well.

### B. Experimental Results for GMM and DNN Classifiers

Inspired by the work in [20], the GMM and DNN models are trained on the $\Delta$ and $\Delta^2$ (first- and second-order frame-to-frame difference, with 40 dimensions) features introduced in Section II. When training GMM-LLR classifiers, the mixture number is set to 512. When training DNN classifiers, a block of 11 frames SBCC/CQCC($\Delta\Delta^2$) are used as the training data. Hence, the number of nodes of the input layer is 440. The output layer has five nodes, the first one is for human speech and the other four are for five known spoofing methods (S3 and S4 have the same label). EER is used for measuring spoofing detection performance. The average EERs of different spoofing features and scoring methods on the evaluation set are shown in Fig. 10.

It can be observed that, among the five SBCC($\Delta\Delta^2$) features, GFCC($\Delta\Delta^2$) and MFCC($\Delta\Delta^2$), which are produced by the filter banks with large spacing in the high-frequency region, perform the worst. IMFCC($\Delta\Delta^2$) and IGFCC($\Delta\Delta^2$),

$$\theta = \frac{\sigma_h^2 m_s - \sigma_s^2 m_h + \sigma_h \sigma_m \sqrt{(m_h - m_s)^2 + 2\left(\sigma_h^2 - \sigma_s^2\right)(\log \sigma_h - \log \sigma_s)}}{\sigma_h^2 - \sigma_s^2} \tag{18}$$

TABLE III
ACCURACIES (AVG. EER IN %) OF DIFFERENT FEATURES AND CLASSIFIERS ON THE EVALUATION SET

| Feature ($\Delta\Delta^2$) | classifier | known | | | | | unknown | | | | | mean | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 | S9 | S10 | known | unknown | all |
| LFCC | GMM-LLR ($S_{GMM}$) | 0.021 | 0.362 | **0.000** | **0.000** | 0.117 | 0.120 | 0.011 | 0.071 | 0.021 | 8.433 | 0.100 | 1.731 | 0.915 |
| MFCC | | 0.011 | 1.478 | **0.000** | **0.000** | 0.364 | 0.299 | 0.022 | 0.038 | 0.021 | 19.52 | 0.371 | 3.980 | 2.175 |
| IMFCC | | 0.071 | 0.572 | 0.005 | **0.000** | 0.266 | 0.298 | 0.071 | 0.641 | 0.138 | 7.582 | 0.183 | 1.746 | 0.965 |
| GFCC | | 0.050 | 1.980 | **0.000** | **0.000** | 0.500 | 0.393 | 0.085 | 0.043 | 0.120 | 25.46 | 0.483 | 5.220 | 2.851 |
| IGFCC | | 0.021 | 0.245 | **0.000** | **0.000** | 0.096 | 0.096 | 0.011 | 0.043 | 0.021 | 7.288 | 0.072 | 1.492 | 0.782 |
| CQCC | | 0.005 | 0.054 | **0.000** | **0.000** | 0.054 | 0.043 | 0.032 | 0.851 | 0.016 | 3.190 | 0.032 | 0.826 | 0.425 |
| LFCC | DNN-HLL ($S3_{DNN}$) | **0.000** | 0.011 | **0.000** | **0.000** | **0.000** | **0.000** | **0.000** | 0.087 | **0.000** | 0.984 | 0.002 | 0.214 | 0.108 |
| MFCC | | **0.000** | 0.060 | **0.000** | **0.000** | 0.011 | 0.011 | **0.000** | **0.000** | 0.005 | 2.658 | 0.014 | 0.535 | 0.274 |
| IMFCC | | **0.000** | **0.000** | **0.000** | **0.000** | **0.000** | **0.000** | **0.000** | 0.174 | **0.000** | 0.734 | **0.000** | 0.182 | 0.091 |
| GFCC | | **0.000** | 0.011 | **0.000** | **0.000** | 0.011 | 0.011 | **0.000** | **0.000** | 0.005 | 4.005 | 0.004 | 0.804 | 0.404 |
| IGFCC | | **0.000** | 0.011 | **0.000** | **0.000** | 0.005 | 0.021 | **0.000** | 0.043 | **0.000** | 2.095 | 0.003 | 0.432 | 0.217 |
| CQCC | | **0.000** | **0.000** | **0.000** | **0.000** | **0.000** | **0.000** | **0.000** | 0.191 | **0.000** | **0.255** | **0.000** | **0.089** | **0.045** |

TABLE IV
ERROR RATE (%) ON EVALUATION SET, USING DNN AS A DECISION DEVICE

| Feature ($\Delta\Delta^2$) | human | mean known (S1-S5) | mean unknown (S6-S9) | S10 | mean (S1-S10) |
|---|---|---|---|---|---|
| LFCC | 0.00 | 3.15 | 10.54 | 96.03 | 15.39 |
| MFCC | 0.00 | 9.16 | **7.43** | 96.10 | 17.16 |
| IMFCC | 0.00 | 1.39 | 9.15 | 94.75 | 13.83 |
| GFCC | 0.00 | 7.22 | 7.26 | 98.45 | 16.36 |
| IGFCC | 0.00 | 4.39 | 9.13 | 95.48 | 15.40 |
| CQCC | 0.00 | **1.21** | 11.97 | **68.41** | **12.24** |

which are generated by filter banks highlighting the high-frequency regions, work better. IGFCC($\Delta\Delta^2$) performs better on the GMM-LLR classifier and IMFCC($\Delta\Delta^2$) better on the DNN classifier which is consistent with the findings in [28].

CQCC($\Delta\Delta^2$) performs the best among all the six investigated features, which indicates that features which have nonuniform resolution on both time and frequency domains can capture the spoofing information more effectively.

Comparing the two different classifiers, the DNN classifier using LLR as spoofing detection scores ($S1_{\text{DNN}}$ and $S2_{\text{DNN}}$), works similar to the GMM-LLR classifiers. However, when using the DNN-HLL scoring method ($S3_{\text{DNN}}$), much better performance can be achieved.

In Table III, we further compare the performance of GMM-LLR and DNN-HLL classifiers, using (1) and (6) as scoring methods, respectively.

It is clearly shown that the DNN-HLL model scoring with the HLL works much better than the GMM-LLR model scoring with the LLR on all the investigated features. Especially on S10 attack, which is very difficult to detect, the DNN-HLL model still works well. The CQCC($\Delta\Delta^2$)-based DNN-HLL classifier can reduce the average EER on unknown attacks to 0.089%. Moreover, the average EER on all attacks is reduced to 0.045%, which is almost 10 times better than the state-of-the-art GMM-LLR classifier. To our knowledge, this is the best performance among all the published results.

When using the trained DNN as a decision device, the error rates for detecting human and different spoofed speech on the evaluation set is shown in Table IV.

It can be observed that when using (3) to make the spoofing decision, about 15% of spoofing speech will be wrongly judged as human speech, on average. For the $S10$ spoofing attack, the error rate can raise to over 60%. This corroborates the conclusion in Section III-C that the scoring methods are more suitable than the decision method for the spoofing detection task.

### C. Analysis of Experimental Results

In Section III-C, we briefly investigate the distribution of human node outputs of the spoofing detection DNN. Here we present detailed results in order to analyze the experimental results in Section IV-B. Histograms of human node outputs generated by different features are shown in Fig. 11.

It can be observed that the output of the human node are concentrated in two intervals, around 0 and 1. About 90% of the outputs from human testing frames fall into the high value range. When using DNN to make the spoofing decisions directly, there is no human speech wrongly rejected as spoofed speech.

On known spoofing attacks (S1–S5), about 80% of the outputs fall into the low value region, and the known/trained spoofed speech can be easily detected. On the unknown attacks S6–S9, which are relatively easy to detect, the ratio of outputs in low value region has been reduced to 60%, and most of the spoofing frames can still be correctly detected.

For S10 attack, histograms show that about 60%–70% of the outputs fall into the high value area, which means most of the spoofing frames are wrongly accepted as human frames. The S10 spoofing samples are produced by concatenating the time-domain waveform of human speech directly without vocoding and feature extraction techniques, which makes the generated frame level features very similar to human speech features. It causes very high error rates in the detection of $S10$ spoofing attacks.

We compute the mean and variance values of human and spoofing scores of speech in the evaluation set. Following (18) and (20), FAR and FRR of different features and scoring methods are shown in Table V.

We can find that when detecting S1–S9, which are relatively easy to distinguish from human speech, the LLR scoring methods ($S1_{\text{DNN}}$ and $S2_{\text{DNN}}$) perform a little better than the

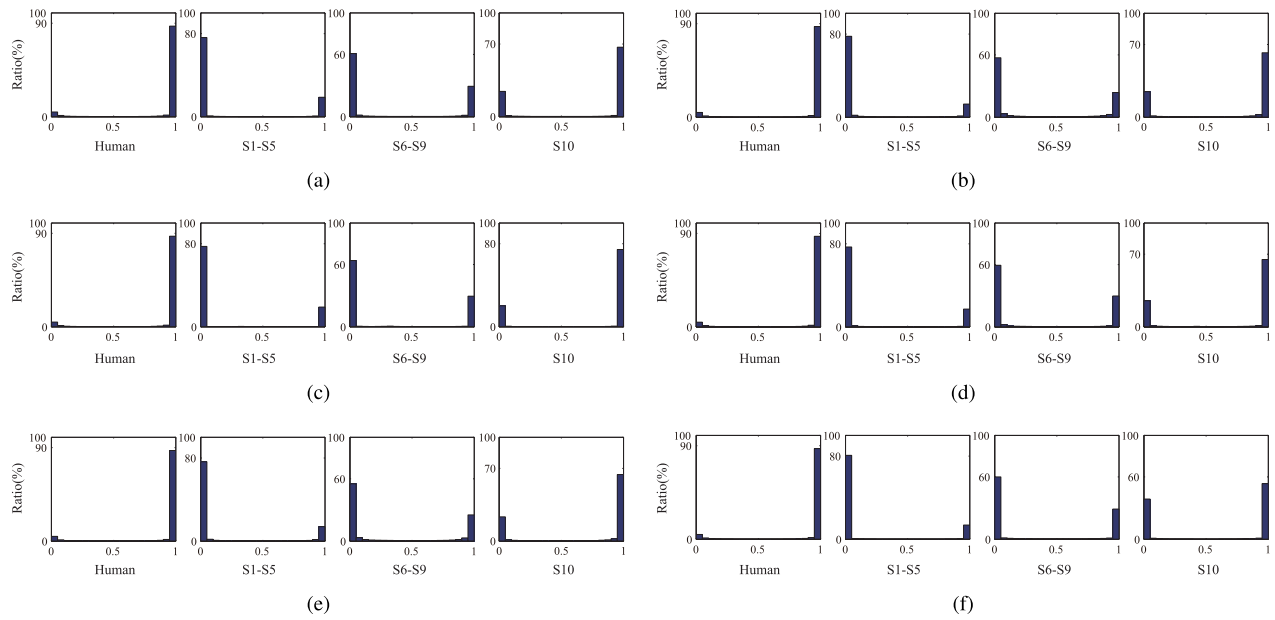



Fig. 11. Histograms of the human node output generated by different features on human speech and spoofed speeches. (a) MFCC. (b) IMFCC. (c) GFCC. (d) IGFCC. (e) LFCC. (f) CQCC.

TABLE V
FRR AND FAR(%) ESTIMATED BY SPOOFING SCORES OF HUMAN/SPOOFING SPEECH IN THE EVALUATION SET

| Scoring method | Feature ($\Delta\Delta^2$) | Evaluation | | | | | |
|---|---|---|---|---|---|---|---|
| | | S1-S5 | | S6-S9 | | S10 | |
| | | FRR | FAR | FRR | FAR | FRR | FAR |
| $S_{GMM}$ | MFCC | 0.31 | 1.66 | 0.46 | 1.65 | 1.73 | 25.74 |
| | IMFCC | 0.31 | 1.09 | 0.52 | 2.58 | 0.73 | 19.75 |
| | GFCC | 0.35 | 1.75 | 0.51 | 1.06 | 2.22 | 29.21 |
| | IGFCC | 0.31 | 0.91 | 0.18 | 0.84 | 1.06 | 16.95 |
| | LFCC | 0.25 | 1.96 | 0.24 | 1.44 | 0.87 | 19.95 |
| | CQCC | 0.27 | 0.16 | 0.71 | 2.65 | 1.57 | 7.39 |
| | mean | 0.30 | 1.13 | 0.44 | 1.70 | 1.36 | 19.83 |
| $S1_{DNN}$ | MFCC | 0.12 | 0.96 | 0.05 | 0.31 | 7.33 | 26.92 |
| | IMFCC | 0.01 | **0.05** | **0.00** | **0.01** | 2.12 | 12.33 |
| | GFCC | 0.14 | 0.72 | 0.01 | 0.02 | 12.55 | 39.39 |
| | IGFCC | 0.15 | 0.76 | 0.02 | 0.03 | 9.81 | 27.31 |
| | LFCC | 0.02 | 0.10 | 0.04 | 0.16 | 4.14 | 19.41 |
| | CQCC | 0.04 | 0.18 | 0.03 | 0.07 | 1.82 | 6.03 |
| | mean | 0.08 | 0.46 | 0.03 | 0.10 | 6.30 | 21.90 |
| $S2_{DNN}$ | MFCC | 0.12 | 0.95 | 0.05 | 0.29 | 6.88 | 25.54 |
| | IMFCC | 0.01 | 0.06 | **0.00** | **0.01** | 1.87 | 11.05 |
| | GFCC | 0.14 | 0.72 | 0.01 | 0.02 | 12.33 | 38.66 |
| | IGFCC | 0.14 | 0.76 | 0.02 | 0.03 | 9.32 | 25.91 |
| | LFCC | 0.01 | 0.10 | 0.03 | 0.13 | 3.56 | 17.28 |
| | CQCC | 0.04 | 0.19 | 0.03 | 0.07 | 1.80 | 5.96 |
| | mean | 0.08 | 0.46 | 0.02 | 0.09 | 5.96 | 20.73 |
| $S3_{DNN}$ | MFCC | 0.03 | 1.69 | 0.04 | 1.21 | 0.66 | 7.95 |
| | IMFCC | **0.00** | 0.11 | **0.00** | 0.03 | 0.21 | 4.25 |
| | GFCC | 0.02 | 1.27 | 0.01 | 0.14 | 0.75 | 10.63 |
| | IGFCC | 0.04 | 1.94 | **0.00** | 0.02 | 0.54 | 5.12 |
| | LFCC | 0.01 | 0.25 | 0.03 | 0.62 | 0.29 | 4.69 |
| | CQCC | 0.01 | 0.73 | 0.02 | 0.36 | **0.10** | **1.58** |
| | mean | 0.02 | 1.00 | 0.02 | 0.40 | 0.42 | 5.70 |

HLL method ($S3_{\text{DNN}}$). For S10 attacks which are very similar to human speech, the HLL method works much better than the LLR method.

It also verifies the conclusion in Section III-C that when the spoofed speech is very different from the human speech the LLR method performs better than the HLL method, while when the spoofed speech is very similar to human speech, the HLL scoring method works much better.

TABLE VI
DESCRIPTION OF JOINT ASV AND SPOOFING DETECTION DATABASE

| Subsets | Development | | Evaluation | |
|---|---|---|---|---|
| | Male | Female | Male | Female |
| genuine | 1498 | 1999 | 4053 | 5351 |
| impostor(Z) | 4275 | 5700 | 8000 | 10400 |
| spoofed(K) | 21375 | 28500 | 40000 | 52000 |
| spoofed(U) | - | - | 40000 | 52000 |

### *D. Experimental Results of Joint Spoofing Detection and ASV System*

The effect of fusing ASV and spoofing detection systems is also evaluated on the ASVspoof 2015 database. The joint ASV and spoofing detection database is described in Table VI. There are two kinds of impostor utterances in the database, zero-effort impostor [Impostor(Z)] and spoofing impostor. Zero-effort impostor utterances still belong to human speech, but the speaker IDs are not enrolled in the ASV system. Spoofing impostor utterances are generated by known [spoofed(K)] and unknown [spoofed(U)] spoofing methods.

Two classical ASV systems: a GMM-UBM [44] system and an i-Vector [3] system are evaluated in this paper. In the two systems, MFCCs with 60 dimensions (static + $\Delta$ + $\Delta^2$) are selected as the ASV features.

In the GMM-UBM system, two gender-dependent UBMs of 512 components are trained with the speech data from the full TIMIT [46] and RSR2015 corpora [47]. Speaker models are generated by maximum-*a posteriori* adaptation with the UBM. ASV scores are calculated by the LLR between speaker models and the UBM.

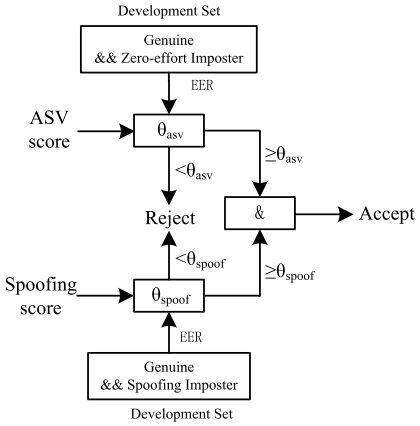



Fig. 12. Fusion of ASV and spoofing detection scores.

In the i-Vector system, gender depended UBMs are also trained by the full TIMIT and RSR2015 databases. When training the total variability matrix **T**, we use the full TIMIT data consisting of 630 speakers (438 male and 192 female) and a subset of 10 different sentences for 300 speakers (157 male and 143 female) from the RSR2015 database. The dimensionality of the total variability subspace is set as 400. LDA is used to reduce the dimension of i-Vectors from 400 to 200 and PLDA is used to compute ASV scores [3]. The same data used for training **T** are also used to train LDA and PLDA. Since each target speaker has five different utterances for enrollment, extracted i-Vectors are averaged to build the speaker models. Finally the ASV scores are calculated by the PLDA similarity between the i-Vector of a test utterance and the claimed speaker model.

The fusion system is shown in Fig. 12. We select the EER threshold, which is obtained from the ASV scores generated by genuine and zero-effort impostor utterances in the development set, as ASV threshold $\theta_{\text{asv}}$. The EER threshold obtained from spoofing scores of genuine and spoofing impostor utterances in the development set is used as the spoofing threshold $\theta_{\text{spoof}}$.

The EER threshold can be calculated following the procedure described in Algorithm 1, where **S** stands for ascending sorted ASV/spoofing scores and **L** means the corresponding labels. $N$ is the number of development utterances, which include $N_T$ true samples (genuine) and $N_F$ false samples [imposter(Z)/spoofed]. The true sample is labeled as 1. By finding the $index$ which can get the minimum difference between $FAR$ and $FRR$, we obtain the EER threshold $\theta$.

The evaluation speech can be accepted only when its ASV score and spoofing score are both larger than $\theta_{\text{asv}}$ and $\theta_{\text{spoof}}$, respectively, otherwise it will be rejected.

FRR and FAR are used to evaluate the performance of the ASV system. The experimental results on evaluation set is shown in Table VII.

When we only use the ASV systems, $\theta_{\text{asv}}$ can distinguish the genuine speech with impostor(Z) utterances. The i-Vector-based system performs better than the GMM-UBM system on defending spoofing attacks. However, the FAR is still significantly increased under known/unknown spoofing attacks, which means that many spoofing impostor utterances are wrongly accepted as genuine utterances.

**Algorithm 1** EER Threshold Computation

**Input:** **S**, **L**, $N$, $N_F$, $N_T$
**Output:** $\theta$
$FRR = 0$; $FAR = 0$;
$n_{fa} = N_F$; $n_{fr} = 0$;
$minDis = +\infty$; $index = 1$
**for** $i = 1 : N$ **do**
  **if** $\mathbf{L}(i) == 1$ **then**
    $n_{fr} = n_{fr} + 1$
  **else**
    $n_{fa} = n_{fa} - 1$
  $FAR = n_{fa}/N_F$
  $FRR = n_{fr}/N_T$
  **if** $abs(FAR - FRR) < minDis$ **then**
    $minDis = abs(FAR - FRR)$
    $index = i$
**return** $\theta = \mathbf{S}(index) - eps$

TABLE VII
PERFORMANCE FOR JOINT ASV AND CQCC-BASED SPOOFING DETECTION SYSTEMS IN TERMS OF FRR AND FAR (%) ON THE EVALUATION SET OF THE ASVSPOOF 2015 DATABASE

| genuine V.S. | Male | | Female | | Male | | Female | |
|---|---|---|---|---|---|---|---|---|
| | FRR | FAR | FRR | FAR | FRR | FAR | FRR | FAR |
| | GMM-UBM | | | | i-Vector | | | |
| impostor(Z) | 7.85 | 6.60 | 7.81 | 6.51 | 9.08 | 18.14 | 9.68 | 9.77 |
| spoofed(K) | 7.85 | 58.78 | 7.81 | 38.47 | 9.08 | 24.22 | 9.68 | 10.83 |
| spoofed(U) | 7.85 | 38.47 | 7.81 | 33.41 | 9.08 | 34.33 | 9.68 | 18.72 |
| | GMM-UBM+GMM-LLR | | | | i-Vector+GMM-LLR | | | |
| impostor(Z) | 7.85 | 6.60 | 7.81 | 6.51 | 9.08 | 18.14 | 9.68 | 9.77 |
| spoofed(K) | 7.85 | 0.20 | 7.81 | 0.01 | 9.08 | **0.00** | 9.68 | **0.00** |
| spoofed(U) | 7.85 | 0.98 | 7.81 | 2.41 | 9.08 | 0.55 | 9.68 | 0.64 |
| | GMM-UBM+DNN-HLL | | | | i-Vector+DNN-HLL | | | |
| impostor(Z) | 7.85 | 6.60 | 7.81 | 6.51 | 9.08 | 18.14 | 9.68 | 9.77 |
| spoofed(K) | 7.85 | **0.00** | 7.81 | **0.00** | 9.08 | **0.00** | 7.81 | **0.00** |
| spoofed(U) | 7.85 | **0.13** | 7.81 | **0.41** | 9.08 | **0.14** | 7.81 | **0.15** |

After combining the ASV system with the spoofing detection system trained on CQCC features, the FRR is not changed, which means that no more genuine utterances are rejected by mistake. The additional spoofing detection system will not affect the acceptance of genuine utterances.

Comparing the two spoofing detection classifiers, the CQCC-based DNN-HLL classifier still works better than the GMM-LLR classifier, especially on resisting unknown spoofing attacks. With the help of the CQCC-based DNN-HLL spoofing detection classifier, by fusing ASV and spoofing scores, most of the spoofing imposter utterances are rejected.

In our experimental setting, the i-Vector system performs worse on detecting imposter (Z) attacks than the GMM-UBM system. However, on resisting spoofing attacks, we obtain a better result by fusing the CQCC-based DNN-HLL classifier with the i-Vector-based ASV system. The FAR can be reduced to 0.00% on known spoofing attacks and to 0.15% on unknown attacks.

## V. CONCLUSION

In this paper, we investigated the performance of GMM and DNN classifiers in spoofing detection. In order to improve

the performance of DNN spoofing detection methods we proposed a novel HLL scoring method, which only uses human node outputs to compute spoofing detection scores. We mathematically proved that, the HLL scoring method is more suitable for the spoofing detection task than the classical log-likelihoods ratio (LLR) method. Five different dynamic filter bank-based cepstral features and constant Q cepstral coefficients (CQCC) were used to train classifiers. The experimental results shown that CQCC features which have variable resolution in both time and frequency domains are more suitable for the spoofing detection task. Comparing the performance between the state-of-the-art GMM-LLR and the newly proposed DNN-HLL method on the ASVspoof2015 database, the DNN-HLL method works nearly 10 times better than the GMM-LLR classifier.

By fusing the DNN-HLL spoofing detection classifier with the GMM-UBM and i-Vector-based ASV systems, the FRR on spoofing attacks can be reduced significantly. In comparison with the two ASV systems, the i-Vector-based system performs better on resisting spoofing attacks. In the future we will investigate the performance of the DNN-HLL model on other kinds of spoofing attacks, such as replay attacks or WaveNet produced spoofing speech [48].

## REFERENCES


[1] D. A. Reynolds, "An overview of automatic speaker recognition technology," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, vol. 4. May 2002, pp. IV-4072–IV-4075.

[2] A. K. Sarkar and Z.-H. Tan, "Text dependent speaker verification using un-supervised HMM-UBM and temporal GMM-UBM," in *Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2016, pp. 425–429.

[3] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 19, no. 4, pp. 788–798, May 2011.

[4] Z. Ma, H. Yu, Z.-H. Tan, and J. Guo, "Text-independent speaker identification using the histogram transform model," *IEEE Access*, vol. 4, pp. 9733–9739, 2017.

[5] N. Li, M.-W. Mak, and J.-T. Chien, "Deep neural network driven mixture of PLDA for robust *i*-vector speaker verification," in *Proc. Spoken Lang. Technol. Workshop (SLT)*, Dec. 2016, pp. 186–191.

[6] Z. Wu, E. S. Chng, and H. Li, "Conditional restricted Boltzmann machine for voice conversion," in *Proc. IEEE China Summit Int. Conf. Signal Inf. Process. (ChinaSIP)*, Jul. 2013, pp. 104–108.

[7] E. Helander, H. Silen, T. Virtanen, and M. Gabbouj, "Voice conversion using dynamic kernel partial least squares regression," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 3, pp. 806–817, Mar. 2012.

[8] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, vol. 1. May 1996, pp. 373–376.

[9] H. Ze, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, May 2013, pp. 7962–7966.

[10] A. Sizov, E. Khoury, T. Kinnunen, Z. Wu, and S. Marcel, "Joint speaker verification and antispoofing in the *i*-vector space," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 4, pp. 821–832, Apr. 2015.

[11] X. Tian, Z. Wu, X. Xiao, E. S. Chng, and H. Li, "Spoofing detection from a feature representation perspective," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Mar. 2016, pp. 2119–2123.

[12] M. Sahidullah *et al.*, "Integrated spoofing countermeasures and automatic speaker verification: An evaluation on ASVspoof 2015," in *Proc. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2016, pp. 1700–1704.

[13] T. Kinnunen, Z. Z. Wu, K. A. Lee, F. Sedlak, E. S. Chng, and H. Li, "Vulnerability of speaker verification systems against voice conversion spoofing attacks: The case of telephone speech," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2012, pp. 4401–4404.

[14] P. L. D. Leon, M. Pucher, J. Yamagishi, I. Hernáez, and I. Saratxaga, "Evaluation of speaker verification security and detection of HMM-based synthetic speech," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 8, pp. 2280–2290, Oct. 2012.

[15] J. Lindberg and M. Blomberg, "Vulnerability in speaker verification—A study of technical impostor techniques," in *Proc. Eurospeech*, vol. 99. 1999, pp. 1211–1214.

[16] Z. Wu, C. E. Siong, and H. Li, "Detecting converted speech and natural speech for anti-spoofing attack in speaker recognition," in *Proc. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2012, pp. 1700–1703.

[17] J. Sanchez, I. Saratxaga, I. Hernáez, E. Navas, D. Erro, and T. Raitio, "Toward a universal synthetic speech spoofing detection using phase information," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 4, pp. 810–820, Apr. 2015.

[18] X. Xiao, X. Tian, S. Du, H. Xu, E. S. Chng, and H. Li, "Spoofing speech detection using high dimensional magnitude and phase features: The NTU approach for ASVspoof 2015 challenge," in *Proc. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2015, pp. 2052–2056.

[19] Z. Wu *et al.*, "Anti-spoofing for text-independent speaker verification: An initial database, comparison of countermeasures, and human performance," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 24, no. 4, pp. 768–783, Apr. 2016.

[20] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A comparison of features for synthetic speech detection," in *Proc. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2015, pp. 2087–2091.

[21] T. B. Patel and H. A. Patil, "Combining evidences from mel cepstral, cochlear filter cepstral and instantaneous frequency features for detection of natural vs. spoofed speech," in *Proc. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2015, pp. 2062–2066.

[22] X. Peng, C. Lu, Z. Yi, and H. Tang, "Connections between nuclear-norm and frobenius-norm-based representations," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2016.2608834.

[23] L. Wang, Y. Yoshida, Y. Kawakami, and S. Nakagawa, "Relative phase information for detecting human speech and spoofed speech," in *Proc. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2015, pp. 2092–2096.

[24] Z. Ou *et al.*, "Utilize signal traces from others? A crowdsourcing perspective of energy saving in cellular data communication," *IEEE Trans. Mobile Comput.*, vol. 14, no. 1, pp. 194–207, Jan. 2015.

[25] C. Hanilçi, T. Kinnunen, M. Sahidullah, and A. Sizov, "Classifiers for synthetic speech detection: A comparison," in *Proc. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2015, pp. 2057–2061.

[26] Y. Liu, Y. Tian, L. He, J. Liu, and M. T. Johnson, "Simultaneous utilization of spectral magnitude and phase information to extract supervectors for speaker verification anti-spoofing," in *Proc. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2015, pp. 2069–2073.

[27] Z. Yang, J. Lei, K. Fan, and Y. Lai, "Keyword extraction by entropy difference between the intrinsic and extrinsic mode," *Phys. A, Stat. Mech. Appl.*, vol. 392, no. 19, pp. 4523–4531, 2013.

[28] H. Yu, A. Sarkar, D. A. L. Thomsen, Z. H. Tan, Z. Ma, and J. Guo, "Effect of multi-condition training and speech enhancement methods on spoofing detection," in *Proc. 1st Int. Workshop Sens., Process. Learn. Intell. Mach. (SPLINE)*, 2016, pp. 1–5.

[29] W. Zuo, D. Ren, D. Zhang, S. Gu, and L. Zhang, "Learning iteration-wise generalized shrinkage–thresholding operators for blind deconvolution," *IEEE Trans. Image Process.*, vol. 25, no. 4, pp. 1751–1764, Apr. 2016.

[30] Y. Deng, F. Bao, Y. Kong, Z. Ren, and Q. Dai, "Deep direct reinforcement learning for financial signal representation and trading," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 653–664, Mar. 2016.

[31] E. Hosseini-Asl, J. M. Zurada, and O. Nasraoui, "Deep learning of part-based representation of data using sparse autoencoders with nonnegativity constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 12, pp. 2486–2498, Dec. 2016.

[32] N. Chen, Y. Qian, H. Dinkel, B. Chen, and K. Yu, "Robust deep feature for spoofing detection—The SJTU system for ASVspoof 2015 challenge," in *Proc. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2015, pp. 2097–2101.

[33] M. J. Alam, P. Kenny, V. Gupta, and T. Stafylakis, "Spoofing detection on the ASVspoof2015 challenge corpus employing deep neural networks," in *Proc. Speaker Language Recognit. Workshop, Odyssey*, 2016, pp. 270–276.

[34] Y. Qian, N. Chen, and K. Yu, "Deep features for automatic spoofing detection," *Speech Commun.*, vol. 85, pp. 43–52, Dec. 2016.

[35] Z. Wu *et al.*, "ASVspoof 2015: The first automatic speaker verification spoofing and countermeasures challenge," *Training*, vol. 10, no. 15, p. 3750, 2015.
[36] M. Todisco, H. Delgado, and N. Evans, "A new feature for automatic speaker verification anti-spoofing: Constant Q cepstral coefficients," in *Proc. Speaker Lang. Recognit. Workshop, Odyssey*, 2016, pp. 249–252.
[37] S. Mika, G. Rätsch, J. Weston, B. Schölkopf, and K.-R. Müller, "Fisher discriminant analysis with kernels," in *Proc. IEEE Signal Process. Soc. Workshop Neural Netw. Signal Process. IX*, Aug. 1999, pp. 41–48.
[38] S. J. D. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. IEEE 11th Int. Conf. Comput. Vis. (ICCV)*, Oct. 2007, pp. 1–8.
[39] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
[40] A. Adiga, M. Magimai, and C. S. Seelamantula, "Gammatone wavelet cepstral coefficients for robust speech recognition," in *Proc. IEEE Int. Conf. Region (TENCON)*, Oct. 2013, pp. 1–4.
[41] X. Valero and F. Alias, "Gammatone cepstral coefficients: Biologically inspired features for non-speech audio classification," *IEEE Trans. Multimedia*, vol. 14, no. 6, pp. 1684–1689, Dec. 2012.
[42] Y. Shao, Z. Jin, D. Wang, and S. Srinivasan, "An auditory-based feature for robust speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2009, pp. 4625–4628.
[43] J. C. Brown, "Calculation of a constant Q spectral transform," *J. Acoust. Soc. Amer.*, vol. 89, no. 1, pp. 425–434, 1991.
[44] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digit. Signal Process.*, vol. 10, nos. 1–3, pp. 19–41, 2000.
[45] D. Yu *et al.*, "An introduction to computational networks and the computational network toolkit," Microsoft, Albuquerque, NM, USA, Tech. Rep. MSR-TR-2014–112, 2014.
[46] J. S. Garofolo *et al.*, "TIMIT acoustic-phonetic continuous speech corpus," *Linguistic Data Consortium*, vol. 10, no. 5, 1993.
[47] A. Larcher, K.-A. Lee, P. L. S. Martínez, T. H. Nguyen, B. Ma, and H. Li, "Extended RSR2015 for text-dependent speaker verification over VHF channel," in *Proc. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2014, pp. 1322–1326.
[48] A. van den Oord *et al.* (Sep. 2016). "WaveNet: A generative model for raw audio." [Online]. Available: https://arxiv.org/abs/1609.03499

**Hong Yu** received the master's degree in signal and information processing from Shandong University, Jinan, China, in 2006. He is currently pursuing the Ph.D. degree with the Beijing University of Posts and Telecommunications, Beijing, China.

From 2006 to 2013, he was a Lecturer with Ludong University, Shandong, China. He has been a Visiting Ph.D. Student with Aalborg University, Aalborg, Denmark, since 2015. His current research interests include pattern recognition and machine learning fundamentals with a focus on applications in image processing, speech processing, data mining, biomedical signal processing, and bioinformatics.

**Zheng-Hua Tan** (M'00–SM'06) received the B.Sc. and M.Sc. degrees in electrical engineering from Hunan University, Changsha, China, in 1990 and 1996, respectively, and the Ph.D. degree in electronic engineering from Shanghai Jiao Tong University, Shanghai, China, in 1999.

In 2001, he joined the Department of Electronic Systems, Aalborg University, Aalborg, Denmark, where he is currently a Professor. He is also a Co-Head of the Center for Acoustic Signal Processing Research, Aalborg University. He was a Visiting Scientist at the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA, an Associate Professor with the Department of Electronic Engineering, Shanghai Jiao Tong University, and a Post-Doctoral Fellow with the Department of Computer Science, Korea Advanced Institute of Science and Technology, Daejeon, South Korea. He has authored or co-authored more than 170 publications in refereed journals and conference proceedings. His current research interests include machine learning, deep learning, pattern recognition, speech and speaker recognition, noise-robust speech processing, multimodal signal processing, and social robotics.

Dr. Tan has served as an Editorial Board Member/Associate Editor for *Computer Speech and Language*, *Digital Signal Processing*, and *Computers and Electrical Engineering*. He was a Lead Guest Editor of the IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING and a Guest Editor of *Neurocomputing*.

**Zhanyu Ma** (S'08–M'11–SM'17) received the Ph.D. degree in electrical engineering from the KTH-Royal Institute of Technology, Stockholm, Sweden, in 2011.

From 2012 to 2013, he was a Post-Doctoral Research Fellow with the School of Electrical Engineering, KTH-Royal Institute of Technology. He has been an Associate Professor with the Beijing University of Posts and Telecommunications, Beijing, China, since 2014. He has also been an Adjunct Associate Professor with Aalborg University, Aalborg, Denmark, since 2015. His current research interests include pattern recognition and machine learning fundamentals with a focus on applications in multimedia signal processing, data mining, biomedical signal processing, and bioinformatics.

**Rainer Martin** (S'86–M'90–SM'01–F'11) received the Dipl.-Ing. degree from RWTH Aachen University, Aachen, Germany, in 1988, the M.S.E.E. degree from the Georgia Institute of Technology, Atlanta, GA, USA, in 1989, and the Dr.-Ing. degree from RWTH Aachen University, in1996.

Since 2003, he has been a Professor of information technology and communication acoustics with Ruhr-Universität Bochum, Bochum, Germany. He has co-authored the book *Digital Speech Transmission Enhancement, Coding and Error Concealment* (Wiley, 2006) along with P. Vary and has co-edited the book *Advances in Digital Speech Transmission* (Wiley, 2008) along with U. Heute and C. Antweiler. His current research interests include signal processing for voice communication systems, hearing instruments, and human machine interfaces.

Mr. Martin is a member of the Speech and Language Processing Technical Committee of the IEEE Signal Processing Society and served as an Associate Editor for the IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING.

**Jun Guo** received the B.E. and M.E. degrees from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 1982 and 1985, respectively, and the Ph.D. degree from Tohuku Gakuin University, Sendai, Japan, in 1993.

He is currently a Professor and the Vice President of BUPT. He has authored over 200 papers in journals and conferences including *Science*, *Scientific Reports* (Nature), the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, *Pattern Recognition*, AAAI, CVPR, ICCV, and SIGIR. His current research interests include pattern recognition theory and application, information retrieval, content-based information security, and bioinformatics.